

AMENDMENTS TO THE SPECIFICATION

Please replace paragraph [0004] with the following amended paragraph:

[0004] Formatted computer files or “formatted files” as referred to herein are available in a variety of different file formats. Execution files are available in the .EXE format. “Script files” are available in such formats as .BAT, .JS, and .VB. Text files are available in the format .TXT and image files are available in such formats as .PDF, .TIF, .JPG, .BMP and .PNG. Audio files are available in such formats as .MP3, .WAV and .AVI and .MID. XML is a computer file format that can support execution files, scripts files, text files, image files, audio files, or combinations of the above types of files. These types of files may be stored in a file system such as Intel’s Persistent Storage Manager (IPSM) System. Certain types of files including dynamic link libraries (~~DLLs~~) DLLs, executable programs, and critical data files can be stored as part of a devices operating system, also referred to as a kernel.

Please replace paragraph [0044] with the following amended paragraph:

[0044] An electrical block diagram of a device, which may be utilized as either a broadcasting device or a receiving device is shown in Fig. 2a. Device 10 may be an optical reader having an image engine including two-dimensional image sensor 32 provided on image sensor chip 132 and associated imaging optics 50. Image sensor chip 132 may be provided in an IT4000 or IT4200 image engine of the type available from ~~HHP, Inc.~~ Hand Held Products, Inc. of Skaneateles Falls, ~~NY~~ New York. Device 10 further includes a processor IC chip 140 such as may be provided by e.g., an INTEL Strong ARM RISC processor, or an INTEL PXA255 Processor. Processor IC chip 140 includes a central processing unit (CPU) 40. Referring to further features of device 10, device 10 may include a display 13d, such as a liquid crystal display, a keyboard 13k, a plurality of radio transceivers such as an 802.11 radio communication link 170, a GSM/GPRS radio communication link 171, a ~~blue-tooth~~ Bluetooth radio communication link ~~470~~ 172. Device 10 may further include Infrared (IR) communication link 175. Keyboard 13k may communicate with IC chip 140 via microcontroller chip 57. Device 10 further includes a memory 45 including a volatile memory and a non-volatile

memory. The volatile memory of device 10 typically is provided in part by a RAM 42. The non-volatile memory may be provided in part by flash ROM 44. Processor IC chip 140 is in communication with RAM 42 and ROM 44 via system bus 60. Processor IC chip 140 and microcontroller chip 57 also include areas of memory 45, volatile and non-volatile. Control circuit 40 of device 10 includes memory 45 and CPU 41. A non-volatile area of memory 45 within microcontroller chip 57 may store a program for operating keyboard 13k. Microcontroller chip 57 may be termed a "keyboard controller."

Please replace paragraph [0045] with the following amended paragraph:

[0045] Processor IC chip 140 may include a number of I/O interfaces not shown in Fig. 2a including several serial interfaces (e.g., general purpose, Ethernet, ~~blue-tooth~~ Bluetooth), and parallel interfaces (e.g., PCMCIA, Compact Flash).

Please replace paragraph [0059] with the following amended paragraph:

[0059] More specifically, a typical boot loader program in portable devices (1) performs functions of a BIOS program, as in a PC, (2) initializes system hardware including IO devices; (3) configures memory, timings, interrupts, and CPU speed; (4) arranges blocks of memory for receipt of a kernel and (5) transfers control of control circuit 140 to a kernel (e.g., DOS, pocket PC, WINDOWS). When a prior art boot loader program is executed, a graphical user interface (GUI) is customarily presented to a user. Typically, at the completion of a boot loader program, a user may be presented by a graphical user interface display screen 1100 on display 13d as presented in Fig. 5, which allows access to a variety of program execution files. Operating in accordance with a graphical user interface supported by an operating system, a pointer device 1102 is made available to a user. A user moves a pointer over an ICON to actuate a selected execution program. Clicking on decode ICON 1104 commences a bar code/OCR algorithm. Clicking on picture taking ICON 1106 commences a picture taking mode of operation in which device 10, on depression of trigger 13t captures an image, and outputs the frame of image data to display 13d or another device without attempting to ~~decide~~ decode decodable symbols therein.

Please replace paragraph [0095] with the following amended paragraph:

[0095] A highly useful alternative embodiment of the invention is described with reference to Figs. 12a-12d. In the embodiment of Fig. 12a, reprogramming symbol 3110 is not printed, but rather is displayed on electronic display 4113d. Reconfigurable device 10c (shown as being provided by a PDA with a detachable imager module such as an ISC-IMAGER or IT4500 Image Reader available from ~~HHP, Inc.~~ Hand Held Products, Inc. of Skaneateles Falls, New York) reads symbol 3110 displayed on display 4113d and, thereby, is reprogrammed. Symbol generator 4110 may display reprogramming symbol 3110 as part of GUI development screen 3600, as shown in the screen shot example of Fig. 7i.

Please replace paragraph [0099] with the following amended paragraph:

[0099] Referring to other aspects of GUI development screen 3600, area 3674 allows a user to select whether data is to be compressed before being encoded in symbol 3110. Compression schemes which may be launched by a user checking the box of area 3674 include e.g., a run-length compression scheme, a Huffman-coding compression scheme, and a Lempel-Ziv compression scheme. Area 3676 allows a user to select whether data encoded in symbol 3110 is to be encrypted prior to being encoded in symbol 3110. Encryption schemes which may be launched by a user checking the box of area 3674 include e.g., an RC2 block cipher scheme, an RC4 stream cipher scheme, a DES block cipher scheme, and a triple DES block cipher. Area 3678 allows a user to select whether device 10c is to be rebooted after executing a command of command 3620. Area 3680 allows a user to select whether a "cycle" command is to be executed. If cycle area 3680 is checked, reconfigurable device 10c waits to read another reprogramming symbol 3110 after reading a first reprogramming symbol 3110. If cycle input area 3680 is not checked, reconfigurable device 10c reverts to main menu display screen 1100 (Fig. 5) after reading symbol 3110. If reboot input area 3678 is checked, device 10c reboots after successfully processing a bar code symbol data stream. If hide batch area 3682 is not checked, device 10c displays during their execution the various commands that make up a script file, if a multiple command script file is entered as the command in command input area 3620 of screen 3600 (see

Examples 4 and 5). Package ID area 3684 allows a user of screen 3600 to input "wild card" data, i.e., any data the user-programmer wishes to enter. Such wild card data may be used for the control of device 10c; that is, processing module 5110 may be made to branch control depending on the content of the data entered in area 3684. When making a reprogramming symbol set as will be described, it is useful to designate each symbol as having the same ID so module 5110 can readily assess whether a decoded symbol is one of a set that is in the process of being read. The package ID data may designate the purpose of a particular symbol 3110, e.g., a song, a printer configuration, a radio configuration, etc. Area ~~3676~~ 3686 indicates to a user the data (e.g. byte) size of the symbol to be encoded. A user may use area 3686 to reduce a symbol size. Symbol generator 110 may be configured so that if a user uses area ~~3386~~ 3686 to reduce a symbol size, symbol generator 110 automatically adjusts a symbol count as indicated by area 3672.

Please replace paragraph [0121] with the following amended paragraph:

[0121] In making a symbol 3110 facilitating a print application, a batch file (.BAT) can be stored on memory 4145 of symbol generator 4110 including a plurality of commands. Such a plurality of commands may include the commands:

TABLE 5A

SAMPLE BATCH FILE
RADIOPOWERUP.EXE /BLUETOOTH
CFGBLUETOOTH.EXE /PRINTER 00.002.72.B0.20.D9 COM8
PRINTREPORT.EXE COM8

Where "RADIOPOWERUP.EXE /BLUETOOTH" is a command which, when executed by device 10c, causes ~~blue tooth~~ Bluetooth radio 172 of device 10c (see Fig. 6b) to be powered up (another radio such as radio 170 could be selected); where "CFGBLUETOOTH.EXE /PRINTER 00.002.72.B0.20.D9 COM8" is a command which causes ~~blue tooth~~ Bluetooth radio 172 to be configured, and where "PRINTREPORT.EXE COM8" is a command which causes printing of a document

selected for printing. Referring to the command "CFGBLUETOOTH.EXE /PRINTER 00.0002.72.B0.20.D9COM8, the argument "00.002.72.b0.20.D9" designates the printer address. The command "PRINTREPORT.EXE COM8" is a command to print a report using communication port COM8. Based on the configuration performed by the previously executed "CFGBLUETOOTH.EXE" command, all COM8 print commands will be routed to printer 4090. The document which is selected for printing may be selected before driving device 10c into a reconfiguration mode in accordance with the invention. Symbol 3110 in Example 5 can be made so that a selected report is printed automatically when symbol 3110 is read. The commands of Table 5A can be stored under a batch file "PRINTERCONNECT.BAT."

Please replace paragraph [0130] with the following amended paragraph:

[0130] Referring to the development screen shown in Fig. 7g, the files FORM.HTM and ~~PROCT.XML~~ PRODUCT.XML are selected using file selection area 3610. No path is selected in area 3650. Thus, decoded data corresponding to the selected file will be stored to the root directory of device 10c. In command entry area 3620, the command "IEXPLORE.EXE /FORM.HTM" is entered. The command IEXPLORE.EXE /FORM.HTM, when executed by device 10c after being decoded, will result in the FORM.HTM file being opened using the browser application program IEXPLORE.EXE.

Please replace paragraph [0138] with the following amended paragraph:

[0138] A symbol 3110 is made which when read by device 10c results in a formatted file being downloaded from a nonintegrated computer device which remote computer (e.g., driver 4052, Fig. 6a) may be using File Transfer Protocol (FTP). In order to make a symbol 3110 which when read results in a file being downloaded into device 10c using FTP, there is no need to encode any formatted file data into symbol 3110. Therefore, no selection need be made in area 3610 of screen 3600. In area 3620, a user inputs a command for encoding into symbol 3110. Referring to screenshot Fig. 7i, the command "TFTP.EXE FTP.HHP.COM/UTILS/UTILS.CAB/IPSM/AUTOINSTALL" is entered in area 3620. Decoding of and execution of the above command by reconfigurable device 10c causes reconfigurable device 10c to install the formatted file

UTILS.CAB on device 10c in directory “/IPSM/AUTOINSTALL” using FTP. Interface 3600 in the example of Fig. 7i includes a displayed time-varying reprogramming symbol 3110TV. In the specific example, the TFTP.EXE utility (Trivial File Transfer Protocol) is used to download the .CAB file UTILS.CAB to the “IPSM/AUTOINSTALL” directory of device 10c from the “/UTILS” directory of the FTP server FTP.HHP.COM. Formatted files can also, in accordance with the invention, be downloaded from remote (~~possible~~ possibly Internet) directories using Hyper Transfer (HTTP) (see Example 2) or another appropriate protocol in the Transmission Control Protocol/internet Protocol (TCP/IP) suite of protocol (e.g., FTP, HTTP, TELNET, SMTP, SLIP, PPP). Skilled artisans will recognize that protocols in the TCP/IP suite can be utilized to download files to portable device 10c from a nonintegrated device that is a common LAN with device 10c (e.g., PC 4110, Fig. 6a), or a device in communication with device 10c via an Intranet or the Internet.